

-1-

EARLY TRAFFIC REGULATION TECHNIQUES TO  
PROTECT AGAINST NETWORK FLOODING

FIELD OF THE INVENTION

5 The present invention is directed to  
communication systems, and more particularly, to flow  
control methods and apparatus suitable for use in network  
congestion control, especially when systems are under  
flooding Denial-of-service attacks.

10 BACKGROUND OF THE INVENTION

Data networks are used today to transmit vast  
amounts of data. Such networks comprise elements  
sometimes called nodes. Nodes may be, e.g., routers,  
15 switches, and/or end-hosts. Among those nodes, routers or  
switches are called network nodes. End-hosts can serve  
as the source or destination of data transmitted through  
a network. In many packet networks, data is transmitted  
between a source and destination device as a flow of  
20 packets. Flows of packets can be categorized by a wide  
range of factors including, e.g., the type of protocol  
used to form and/or transmit the packet and/or the  
specific type of application to which the packet  
corresponds.

25

EXPRESS MAIL # EL87256022245

As known in the art, it is common to monitor traffic flows and store flow statistics in a database, e.g., for purposes of load balancing and traffic route determination. Gathered traffic information for a node  
5 typically includes information such as packet flow rates and, for each flow, protocol type, application type, source IP address, source port number, destination IP address, destination port number, etc. Such detailed statistics along with information about the time periods  
10 in which such statistics are gathered can be used to group traffic flows into a wide number of classes depending on the intended purpose of grouping the traffic.

15 Flooding Network DoS (N-DoS) attacks occur in a network when one or more sources send large amounts of data to a destination node, e.g., web page server, in an attempt to interfere with the normal servicing of traffic at the destination node. Flows of traffic used to  
20 implement N-DoS attack can be considered malicious since their purpose is to interfere with the communication and servicing of legitimate network traffic.

25 Malicious flows associated with an flooding N-DoS attack often create congestion at certain nodes located prior to, i.e., upstream from, the flow's destination node. The nodes at which congestion occurs are sometimes referred to as bottleneck nodes.

As a result of malicious sources flooding a bottleneck node with traffic, legitimate traffic passing through the bottleneck node may be subject to dropping of packets thereby preventing legitimate communications. Thus, N-DoS attacks negatively effect legitimate users, and/or even cause its victim's services (e.g. web sites) to crash due to excessive loading.

One known technique for protecting against N-DoS attacks involves explicit signature capture and analysis. For example, those signatures can be communication port numbers, daemon names or commands, or contained in IP packet payload. Unfortunately these approaches can be ineffective and may result in negative consequences for legitimate users, because the signatures can change over time making a captured signature useless in identifying a malicious source during a subsequent attack.

Another disadvantage of the signature capture system is that the signature collection methods are an aftermath defense approach. Thus, such an approach helps in preventing future attacks with known signatures, but is of limited use during initial attacks.

In view of the above discussion, it is apparent that there is a need for methods of effectively identifying malicious traffic flows, e.g., traffic flows

from individuals and/or sources involved in launching an N-DoS attack. There is also a need for methods and apparatus for reducing and/or eliminating the effects of malicious traffic flows associated with N-DoS attacks. It is desirable that at least some congestion control methods be capable of limiting malicious traffic prior to a significant collapse or restriction on legitimate network traffic occurs.

10 SUMMARY OF THE INVENTION

The present invention is directed to congestion control methods and apparatus. Various methods and apparatus of the invention are well suited for defending against flooding network Denial-of-Service (N-DoS) attacks.

An Anti-Flooding Flow-Control (AFFC) mechanism of the present invention monitors, analyzes, and regulates traffic flows at network nodes, e.g., routers, based on the flow's behavior. In a node, the AFFC mechanism of the invention, utilizes a traffic baseline generating module, a dynamic buffer manager module, a packet scheduler module, and optionally, an early traffic regulator (ETR) module. Each module may be implemented using software and/or hardware.

In some embodiments traffic baselines are generated external to a node using traffic information

for the particular node. The generated baselines are then supplied to the dynamic buffer manager and packet scheduler in the node. In such embodiments, the traffic baseline module may be implemented as a stand-alone  
5 device separate from packet forwarding nodes. This can reduce the processing burden placed on such nodes by the AFFC methods of the invention.

While the AFFC mechanism can be implemented in  
10 a single node, for more effective network protection it can be implemented in multiple network nodes. AFFC modules, e.g., ETR modules, of different nodes may, and in various embodiments do, interact with one another to perform a multi-node approach to congestion control.

15 The traffic baseline generating module receives and analyzes traffic statistics to generate baseline flow statistics, e.g., diurnal flow statistics, for individual periods of time, e.g., hours or minutes of a day in a  
20 week. The traffic baselines are generated for each node based on the traffic through the node over an extended period of time, e.g., multiple weeks.

As part of the flow control method, the current  
25 data flow rates are compared to the corresponding baseline flow rate for the same period of time and type of traffic. Flows are determined to be aggressive if they have an arrival rate that is higher than the baseline for flow of its type. In accordance with the

present invention, under certain circumstances aggressive flows are targeted for forced data rate reductions. In addition to aggressive flows, unresponsive elastic flows may be blocked independently of traffic baselines.

5

The dynamic buffer manager module 224 and packet scheduler module 226 are the mechanisms by which forced reductions in data flow rates are implemented at a node in response to the presence of congestion. In accordance with the invention the forced data flow reduction functionality of the buffer manager and packet scheduler normally remain inactive. However, when congestion is detected or a control message is received from another network node as part of the ETR method of the invention, the forced data flow reduction functionality in a node is activated. An ETR message triggering activation of the buffer manager and packet scheduler functionality may be received from, e.g., a downstream node confronting a potential collapse due to congestion.

The dynamic buffer manager module 224 of the invention determines packet dropping rates to be applied to different data flows, e.g., those flows identified as being allowable but aggressive. The packet scheduler module 226 determines current packet forwarding rates, e.g., flow rates.

During periods of congestion during which the forced data flow reduction is applied, incoming data flows are processed based on their traffic types, elastic traffic and best effort traffic. Elastic traffic, which  
5 is not responsive to congestion signaling, e.g., ECN (Explicit Congestion Notification) or packet dropping, is considered malicious and dropped.

Elastic traffic that is responsive to  
10 congestion signals is considered allowable.

For both elastic traffic and best-effort traffic, allowable traffic flows are determined to be aggressive if the flow rate of the allowable flow exceeds  
15 a corresponding baseline flow rate. Allowable non aggressive flows, e.g., flows having a flow rate equal to or lower than a corresponding baseline flow rate are forwarded without being subject to flow rate reduction. Allowable flows that are found to be aggressive, are  
20 subject to forced reductions in their flow rates during periods of congestion. The applied flow rate reduction may, e.g., reduce the flow rate of an aggressive flow, to or below the corresponding flow rate baseline.

25 To support different packet drop rates for each allowable aggressive flow, packets from different allowable aggressive flows are stored in different packet forwarding queues. e.g., one per allowable aggressive flow. In some embodiments, e.g., where sufficient memory

is not available to support one queue per flow, a group of flows (e.g. from the same domain) may be processed per queue.

5           The dynamic buffer manager module 224 of the invention determines packet dropping rates to be applied to different data flows, e.g., those flows identified as being allowable but aggressive. The packet scheduler module 226 determines current packet forwarding rates,  
10 e.g., flow rates. As mentioned above, the current flow rates are compared to the baseline flow rates and packets are dropped, e.g., when the current flow rate exceeds the baseline flow rate. Accordingly, incoming flows are subject to different reductions in their flow rates as a  
15 function of their normal baselines and their current arrival rates. In the case of malicious traffic flows, such forced data rate reductions may be interpreted as punishing of the malicious flows.

20           ETR is a mechanism by which congestion control, and forced data rate reductions can be triggered in nodes upstream of a bottleneck node where the congestion occurs. ETR messages are used to activate flow reduction in the upstream nodes. Thus ETR offers protection for  
25 downstream nodes facing potential collapse due to congestion by reducing the flow of traffic directed to the node suffering from congestion.



Numerous additional features and advantages of the invention are discussed in the detailed description which follows.

5 BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 illustrates a communications system incorporating nodes that implement the present invention.

10 Fig. 2 illustrates an exemplary router implemented in accordance with the present invention that may be used as one of the routers shown in Figure 1.

15 Fig. 3 illustrates the steps of an exemplary traffic baseline generation routine of the invention.

Fig. 4 illustrates an exemplary flow baseline table generated and used in accordance with an embodiment of the present invention.

20 Fig. 5 illustrates the steps of an Anti-Flooding Flow-Control (AFFC) method implemented in accordance with an exemplary embodiment of the present invention.

25 Fig. 6 illustrates an exemplary set of internet traffic statistics measured right during a period of potential congestion collapse at a bottleneck node.

Fig. 7 illustrates an exemplary set of router throughput statistics resulting from the AFFC method of the invention being applied at a bottleneck node to the flows listed in Figure 6.

5

Fig. 8 illustrates the dropping of packets from a queue in accordance with the invention.

Fig. 9 illustrates an early traffic regulation method of the invention.

10

Fig. 10A and 10B illustrate early traffic regulation modules implemented in accordance with the invention.

15

Figs. 11 and 12 illustrate signaling between various nodes performed in accordance with the invention.

#### DETAILED DESCRIPTION

20

The present invention is directed to congestion control methods and apparatus. The methods and apparatus of the present invention are well suited for defending against network Denial-of-Service (N-DoS) attacks.

25

Figure 1 illustrates a communications system 100 implemented in accordance with the present invention. The system 100 comprises a plurality of sources 102, 104, 106, an internet 108 and a plurality of destination nodes

110, 112, 114. The internet 108 may be a corporate internet or the world wide Internet. The internet 108 comprises a plurality of nodes R1 through R10 116, 118, 120, 122, 124, 126, 127, 128, 130, 132 connected together  
5 as shown in Figure 1 by the use of solid lines. Each of the nodes may be, e.g., a router or a switch. Arrows are used in Figure 1 to indicate the flow of packets, e.g., between source devices S1, S2, ..., SN, 102, 104, 106 and destination device 112. While Figure 1 shows flows of  
10 packets to destination device D2 112 from sources S1, S2, ..., SN, 102, 104, 106 the communications paths in the system 100 between the routers and devices are bi-directional allowing for responses, e.g., packets and messages, to be transmitted in the reverse direction as  
15 well. In the Figure 1 embodiment source S1 102 is coupled to the internet 108 by router R1 116. In addition, source S2 is coupled to the internet 108 by router R4 122, while source SN 106 is coupled to the internet 108 by router R8 128. Router R7 127 couples  
20 each of the three destination devices, D1 110, D2 112, and D3 114, to the internet 108. As a result packets from any one of the sources 102, 104, 106 will pass through router R7 prior to reaching one of the destination devices 110, 112, 114.

25

Since traffic directed to a destination device, e.g., device D2 112, will pass through the router R7 127 regardless of the source of the traffic, router R7 127 represents a potential congestion point. For purposes of

explaining the invention, router R7 127 will be referred to as a "bottleneck" node since it is the point in system 100 where traffic bottlenecks may occur when excessive amounts of traffic are directed to one of the destination devices D1, D2, D3 110, 112, 114. Relative to the bottleneck node 127, system elements, e.g., routers and sources located to the left of bottleneck node 127, are upstream nodes. System elements, e.g., devices D1, D2, D3 110, 112, 114 to the right of the bottleneck node 127, are downstream nodes since they follow bottleneck node 127 in terms of the exemplary traffic flows shown in Figure 1.

A flooding N-DoS attack works by transmitting a large amount of traffic from one or more sources to a particular destination, e.g., a web page server, which is the target of the N-DoS attack. For purposes of explaining the invention, when discussing an exemplary N-DoS attack it will be assumed that destination device D2 112 is the target of the exemplary attack. The exemplary N-DoS attack will result in bottleneck node 127 being flooded with useless information, corresponding to malicious data flows associated with the N-DoS attack.

Figure 2 illustrates an exemplary router 200 implemented according to one embodiment of the present invention. The router 200 may be used as, any one of the routers shown in Fig. 1 including edge router 127 which serves for discussion purposes as the exemplary

bottleneck node. Router 200 comprises of a CPU 202, a packet forwarding engine 204, an I/O interface 208 and a memory 210. These elements are coupled together by bus 206.

5

The CPU 202 controls operation of the router 200 under direction of various routines stored in memory 210. The packet forwarding engine 204 is responsible for controlling the forwarding of packets under direction of various routines executed by the CPU 202. As part of the forwarding process packet forwarding engine 204 may store received packets corresponding to different flows and/or classes in different queues. In accordance with the congestion control mechanisms of the present invention some received packets may be dropped by the packet forwarding engine, e.g., when the router 200 is incapable of forwarding packets at the rate at which they are received. The I/O interface 208 couples the router to other routers and/or host devices, e.g., source and/or destination devices. Thus, via I/O interface 208 the router 200 receives and transmits packets.

Memory 210 includes a traffic monitoring routine 216, traffic classifier 218, forwarding and flow control routine 220, traffic baseline generating module 222, dynamic buffer manage module 224, packet schedule module 226, early traffic regulator (ETR) module 228, traffic statistics 230, traffic baselines 232, a Recycling table 214, and a plurality of class based

packet queues 234. The traffic statistics 230 include  
current traffic statistics 231 and long term, e.g.,  
weekly traffic statistics 233. The various modules 222,  
224, 226, 228 may be implemented as, e.g., software  
5 routines. Alternatively, the modules could be  
implemented in hardware, e.g., in the router 200, to  
enhance processing speed.

The traffic monitoring routine 216 monitors to  
10 detect when the rate at which packets are received  
exceeds the maximum forwarding packet rate of the router  
200 for a period of time corresponding to the buffering  
capacity of the router 200. This condition corresponds  
to saturation at the router 200 necessitating the  
15 dropping of some received packets. The traffic  
monitoring routine 216 notifies the forwarding and flow  
control routine 220 when saturation occurs and the length  
of the period of saturation.

20 The traffic classifier 218 is used to classify  
packets into different classes and/or flows of traffic  
based on such factors as source address, destination  
address, protocol type, and application type. The  
application type is determined from the port number or  
25 message type information included in a packet. The  
level, e.g., resolution, of traffic classification  
applied by the classifier 218 may depend on the  
application for which the classifier is being used. The  
traffic classifier can be called by the traffic baseline

generating module 222 and/or the forwarding and flow control routine 220 during normal operation.

Forwarding and flow control routine 220 is  
5 responsible for controlling the forwarding and flow control of packets in accordance with the present invention. The forwarding and flow control routine 220 is responsible for activating the dynamic buffer manager module 224, packet scheduler module 226 and early traffic  
10 regulator module 228 used to implemented forced reductions in flow data rates in accordance with the present invention when the router 200 is saturated by packet traffic, e.g., for a preselected period of time. By limiting implementation of all or some of the flow  
15 reduction features of the present invention until saturation of a node has occurred for a period of time, application of the anti-flooding flow reduction techniques can be limited to cases where flooding is likely to be occurring or to cases where a traffic  
20 collapse at the node is likely to occur if steps are not taken to avoid the collapse. In addition, false alarms of which can be induced by occasional short-term traffic spikes can be reduced or eliminated.

25 Traffic baseline generating module 222 operates in parallel with the forwarding and flow control routine 220 to generate traffic baselines for various traffic flows at preselected intervals. As will be discussed below, the traffic baselines 232 are generated from the

traffic statistics 230 which are produced over an extend  
period of time, e.g., days or weeks. Traffic baselines  
232 are stored in memory for use in forced traffic flow  
reduction operations implemented in the router 200 in  
5 accordance with the invention.

The dynamic buffer manager module 224 and  
packet scheduler module 226 implement in accordance with  
the invention are responsible for performing forced  
10 reductions in the rate of packet flows through the router  
200. The amount of the flow reduction applied to  
individual flows or flow groups is determined as a  
function of the traffic baselines 232.

15 Early traffic regulator module 232 is used to  
send early traffic regulation (ETR) signals, e.g.,  
messages, to upstream nodes to trigger the congestion  
control and forced packet flow reductions techniques of  
the present invention to be implemented in the upstream  
20 node. In the case of a node receiving an ETR message,  
the ETR module 228 is responsible for responding to the  
ETR message by implementing forced packet flow rate  
reductions. In some embodiments the forced packet flow  
rate reductions in response to an ETR message are on  
25 flows directed to the node which was the source of the  
ETR message while in other embodiments, the forced packet  
flow rate reductions are limited to flows destined for  
target IP address(es) identified in the received ETR  
message.



An exemplary traffic baseline generating routine 300, which can be used as the traffic base line generating module 222, is shown in Fig. 3.

5

Traffic baseline generation involves removing spikes in a time window corresponding to a daily period of time and removing short-term abnormal traffic patterns which may be encountered over a period of weeks.

10 Furthermore, depending on the implementation, traffic classification for baseline generation purposes need not be limited to simply protocol types and IP addresses but may also be based on the application type and/or port numbers associated with packets.

15

The routine 300 starts in step 301 wherein the routine is executed by the CPU 202. Then in step 302, packets 325 for a current time period  $\Delta T$  are received. The time period  $\Delta T$  may be fixed or any desired length  
20 depending on the environment and specific traffic patterns and/or day and time of week for which the flow rate baseline is being generated. The packets 325 correspond to multiple destination addresses and sources. Accordingly, many flows of traffic are represented in the  
25 received packet stream 325. Each packet includes source IP addresses, destination IP addresses, protocol type information, port number and/or message type information. As discussed above, port number and/or message type information can be used to identify an application to

which the packet corresponds.  $\Delta T$  in the exemplary embodiment is a time segment, e.g., 30 minute time segment, from a 24 hour time period.  $\Delta T$  can be either constant through the day, e.g., 30 minutes, or vary at different time of the day, e.g., 120 minutes from 2 A.M. to 4 A.M. vs. 30 minutes for other busier times of the day. Holidays may be treated as special cases for purposes of data collection and baseline generation if desired, e.g., due to very different traffic patterns on those days.

In step 303, each packet is classified based on destination address, protocol type and application type. The resulting sets of data correspond to individual classes of traffic.

Some exemplary protocol types are TCP and UDP, while some exemplary applications that use TCP are web traffic and FTP traffic. An exemplary application that uses UDP is Echo traffic. Accordingly, the first exemplary traffic class might include TCP packets for an FTP application directed to a first destination. A second exemplary traffic class might include TCP packets for a web application directed to the same destination. Numerous other traffic classes are possible with the granularity of the class being a matter of design choice.

Each class will normally include multiple flows with the number of bits received in each flow varying as

a function of time. The number of bits received in each second by each flow in a class is used in steps 304, 305, and 306, which operate in parallel.

5           In step 304, a sum of the maximum number of bits received from any one flow during each second of time period  $\Delta T$  is generated. The running maximum sum, tends to be larger than the number of bits received in any one of the flows since it may be generated from  
10 values obtained from different flows.

          In step 305, a sum of the total bits received by the node for all the flows in the class being processed during time  $\Delta T$  is generated.

15           In step 306, a sum of the minimum number of bits received from any one flow during each second of time period  $\Delta T$  is generated. This running minimum sum tends to be smaller than the minimum number of bits  
20 received in any one of the flows since it may be generated from values obtained from different flows.

          The running sums of max, min and total bits received are stored in memory elements 235, 237, 239  
25 during processing.

          Once the maximum, total, and minimum number of bits are determined for time period  $\Delta T$  in steps 304, 305,

306 operation proceeds to step 307. In step 307, the max and min sums are subtracted from the total number of bits to generate a smoothed total sum.

5 In step 308, the smoothed total sum is divided by the number of seconds in the time period  $\Delta T$  and the number of flows in the class being processed minus 2. The subtraction of 2 from the total number of flows represents the elimination of bits corresponding to the  
10 composite min and max data flows used to reduce the impact of transient or abnormal flow characteristics. The result of the division operation is a smoothed average value, the current average flow data rate.

15 The current average flow data rate is then stored in step 310 thereby updating the set of long term, e.g., multi-week, traffic statistics 233.

The traffic baseline generating method 300  
20 continues in step 312 by retrieving stored average flow data rates for the corresponding time periods  $\Delta T$  generated from statistics 233 for the preceding weeks. For example, average flow rates may be retrieved in step 312 from the long term traffic statistics 233 for the  
25 four weeks preceding the current time period.

Once again, to reduce the risk of flow rate anomalies having a large impact on a class baseline, in step 314, the minimum and maximum average flow rates,

from the set of average flow rates including the average flow rates for the preceding weeks and the current week, are excluded from the set of data used for generating the class flow rate baseline. Assuming the flow rates for 4  
5 preceding weeks were retrieved in step 312, excluding the minimum and maximum average weekly flow rates will result in 3 average weekly flow rates remaining for the traffic class being processed.

10 In step 316 the flow rate baseline for the class of traffic being processed is generated by averaging the remaining average weekly flow rates. Next, in step 318 the generated flow rate baseline for the traffic class being processed is stored, e.g., in the set  
15 of baseline statistics 232. The most recent class flow baseline for a particular time period may replace a previously generated baseline for the same time period thereby ensuring that the set of traffic baselines 232 includes the most recently generated baseline flow rates  
20 for the various classes of traffic and time periods of interest.

In the above described manner, flow rate baselines are generated for a class based on packet  
25 arrival rates. Alternatively, such baselines could be generated from other flow statistics, e.g., packet forwarding rates.

After flow rate baselines are generated for each class of traffic for which flow rate information is received, the flow rate baseline generation process stops in step 320 pending receipt of another set of flow  
5 packets for processing in the next time period  $\Delta T'$ .

While baselines are described as being generated in the above described manner for classes defined by destination address, protocol type, and  
10 application type, classes may be defined with less granularity, e.g., without considering application type or port number. In such cases, flow rate baselines for the classes can be generated by applying steps 304, through 318 to the traffic corresponding to each class as  
15 defined for the particular application.

Figure 4 illustrates an exemplary flow baseline table 400 for a single destination address. The traffic  
baselines 232 include information of the type shown in  
20 Fig. 4, for each destination address of interest, e.g., each downstream web server likely to be the target of a N-Dos attack.

The table 400 includes base lines for four  
25 different classes of packet traffic corresponding to a single destination address D2. In Fig. 4, each column 402, 404, 406, 408 corresponds to a different traffic class as defined, for the given destination address, by the protocol type (TCP or UDP) and application type (Web,

FTP, Echo, or DNS), as determined by port number and/or message type. The second from the last line of the table 400 indicates the baseline flow rate for each class. Directly below each baseline flow rate is listed the  
5 class to which it applies.

In the Fig. 4 example, class 1 which corresponds to Web traffic has a baseline flow rate of 1000 bits/s per flow. Class 2 which corresponds to FTP  
10 traffic has a baseline flow rate of 500 bits/s. In addition class 3 which corresponds to Echo traffic has a baseline flow rate of 200 bits/s while class 4 which corresponds to DNS traffic has a baseline flow rate of 100 bits/sec.

15 The packet forwarding and flow rate control routine of the present invention will now be described with reference to Fig. 5. Fig. 5 illustrates an exemplary forwarding and flow rate control routine.

20 The routine 220 begins in step 502 when the router which includes the routine is activated and the routine is executed. Once the routine 220 is activated, control step 504 is performed on an ongoing basis.  
25 Control step 504 involves making a determination as to whether the router is currently experiencing congestion sufficient to merit the application of the AFFC mechanisms supported by the dynamic buffer manager module 224, packet scheduler module 226 and, optionally, the

early traffic regulator module 228. In step 504,  
congestion sufficient to merit application of the AFFC  
flow control mechanisms is declared when the router  
encounters a saturation condition which persists for a  
5 preselected period of time, e.g., a period of time  
indicative of persistent and potentially hostile volumes  
of traffic.

In one particular embodiment, the congestion  
10 decision of step 504 is made based on two conditions to  
reduce false positives, the first condition is that the  
summation of total bandwidth shares at the node must  
saturate the node and second, the saturation condition  
must persist for a window period after the saturation  
15 condition initially occurs. In such an embodiment,  
congestion is declared in step 504 when the two  
conditions are met. Congestion is no longer found to be  
present when the saturation condition ceases to be  
encountered for the set period of time in which it was  
20 required to occur before congestion was declared.

When control step 504 determines that  
congestion does not exist, operation proceeds directly to  
step 527 with the received packets being forwarded  
25 without being subject to processing in accordance with  
the invention to reduce the flow rates of specific  
identified traffic flows.



However, when control step 504 detects the presence of congestion, e.g., the two conditions above are satisfied, traffic is processed along the path which proceeds to step 508 via optional step 506.

5

In optional step 506, ETR signaling is initiated. Such signaling will, as discussed below, trigger traffic regulation and flow control to be applied at an upstream node thereby reducing the traffic burden on the current node. When ETR is not used traffic flow processing proceeds directly from step 504 to step 508.

In step 508, the traffic classifier 218 is used to classify the individual incoming traffic flows to corresponding classes. The classes correspond to either elastic traffic, e.g., TCP traffic, or best effort traffic, e.g., UDP traffic. Processing then proceeds to step 510 which represents a fork for the processing of flows falling into either elastic or best effort classes.

20

Individual elastic flows, e.g., TCP flows, proceed from processing fork 510 to step 512 wherein each individual elastic flow is analyzed to determine if it is responsive to congestion signals, e.g., ECN or packet dropping.

25

Due to the congestion control avoidance scheme implemented in elastic protocol such as TCP, elastic traffic will normally decrease its packet send rate in

response to congestion signals. This responsiveness will be reflected, and can be detected, by a decrease in the arrival rate of packets from the source at the nodes through which the traffic flows.

5

In the case of N-Dos attacks, due to the use of spoofed IP addresses, some malicious sources may not respond to congestion signals. Other malicious sources without IP-spoofing may intentionally ignore congestion signals. despite an ever increasing number of dropped packets. Such traffic flows are identified in step 512 as non-responsive, e.g., from their arrival rates which are higher than expected decreasing rate.

15

In one exemplary embodiment detection of responsiveness of a flow is derived from a specific TCP-friendly-traffic behavior, which is bound to a definite end-to-end congestion control and avoidance scheme -

$$T \leq \frac{1.5\sqrt{2/3} * B}{R * \sqrt{p}}; \text{ where } T \text{ is the maximum sending rate, } B \text{ is the}$$

20 number of packet sent,  $R$  is the round-trip-time, and  $p$  is the packet dropping-rate.

In the exemplary embodiment the above behavior is checked in step 512 by measuring the relationship of packet-dropping rate  $p$  and packet arrival rate  $T$ . In the case of responsive TCP traffic when a packet dropping rate increases to a factor of  $\sqrt{x}$ , the packet arrival rates should decrease by a factor of  $a * \sqrt{x}$ ; where  $a$  is an

25

adjusting argument in considering link saturation situation and where  $a$  is usually  $a \leq 1$ . Failure to detect the expected decrease in packet arrival rates results in a determination that a flow is non-responsive.

5

The recycling table 214 is used to record responsive elastic flow information for each class. The information includes data such as <source IP address>, <time-stamp>. Oldest flow records may be replaced by the newest record when the table is full. The table size may be restricted the amount of memory the network node can afford to allocate to this use. The responsiveness of an elastic flow in the table may be checked every time the next packet of the same flow arrives.

15

Processing of non-responsive flows proceeds from step 512 to step 520 wherein the non-responsive flows are blocked. The processing of the received packets corresponding to a non-responsive flow then stops in step 528.

If in step 512, an elastic traffic flow is determined to be responsive to congestion signaling, the flow is considered allowable and flow processing operation proceeds to step 516.

25

Best effort traffic is not expected to vary its data rate in response to the dropping of packets. For this reason, best effort traffic is not analyzed for

responsiveness and is considered to be allowable traffic. Accordingly, for flows determined to correspond to best effort traffic, operation proceeds directly from step 510 to step 516.

5

Each of the allowable traffic flows or flow-groups is compared to the corresponding baseline for a flow of the same class and time, e.g., as indicated by the time and day of the week.

10

Allowable flows, i.e., best effort and responsive elastic traffic flows, having a flow rate equal to, or less than the corresponding baseline flow rate to which they are compared, are categorized as non-aggressive. Processing of non-aggressive flows proceeds to step 527 wherein the packets are forwarded.

15

However, allowable flows which have current flow rates exceeding the corresponding baseline flow rates are categorized as aggressive and identified for forced reductions in their flow rates. Processing of aggressive allowable flows proceeds from step 516 to step 518 wherein the forwarding rates of the aggressive flows are regulated, e.g., by dropping packets from the aggressive flows.

20

25

In step 526, the packet forwarding rates of each aggressive flow is regulated separately as a function of the flow's determined current flow rate and

the corresponding baseline flow rate. Forced reduction in a flow's forwarding rate is implemented by adjusting the maximum threshold 802 of the queue 800 of a flow or flow group as shown in Figure 8. The forced flow forwarding rate reduction is achieved, in one embodiment of the invention, by dropping the required number of received packets from the aggressive flows packet forwarding queue. The drop rate, e.g., penalization severity, for each aggressive flow is affected by the packet arrival rate of the flow. The higher the packet arrival rate of the flow above the baseline flow rate, the higher the applied packet drop rate will be.

Fig. 8 illustrates a flow rate reduction technique which can be applied in step 518 to allowable aggressive flows in accordance with the invention to control packet drop rates as applied to a packet queue 800. As discussed above the packets corresponding to individual aggressive flows are stored in different packet queues. In Fig. 8 an exemplary flow rate reduction technique uses two different thresholds, which are queue fullness thresholds to control packet drop rates, a min threshold 804 and max threshold 806.

In the Fig. 8 example, the packet dropping-rate  $\nu$  is affected by the packet forwarding-rate  $\lambda_{out}$ , which in turn is bound to the class baselines.

The relationship of the packet dropping rate and the maximum threshold is presented in Formula 1.

$$new\ max\ Thresh(t) = \frac{[x(t)-1] * \min\ Thresh(t-\Delta t) + \max\ Thresh(t-\Delta t)}{x(t)} \dots\dots\dots 1$$

Where the  $x(t)$  is the change factor of packet dropping rate  $\nu(t)$  within time frame  $\Delta t$ :

$$x(t) = \frac{\nu(t)}{\nu(t-\Delta t)} .$$

In step 518, the forwarding rates are regulated by a penalty factor  $k$ , which may be preset, e.g., to two

10 --  $\lambda(t) = \frac{\lambda(t-\Delta t)}{k}$ . This calculation can be implemented using a shift operation with little cost. In the above equation  $\lambda(t)$  is the packet forwarding rate at time  $(t)$  while  $\lambda(t-\Delta t)$  is the forwarding rate at time  $(t-\Delta t)$ .

15 After flow rate reduction is applied to the aggressive traffic flows, the remaining packets are forwarded in step 529. Processing of a set of received packets then steps in step 530.

20 Thus, in accordance with the present invention, packets corresponding to elastic non-responsive traffic will be blocked, i.e., it will not be forwarded. In addition, packets corresponding to elastic responsive flows and best effort flows will be passed subject to

25 forced reductions in their flow rates when the flows

exceed the baseline flow rates for flows of the same class of traffic.

Notably, in accordance with the invention the  
5 more aggressive the traffic flow, the greater the applied  
reduction in flow forwarding rate. Accordingly, while  
flows having flow rates within the normal baseline will  
be forwarded with less constraint during periods of  
congestion, flows with higher flow rates will be  
10 subjected to higher forwarding rate reduction. Thus, the  
flows corresponding to a N-DoS attack are more likely to  
be penalized do to their distinct traffic behavior which  
differs from that of legitimate normal flows.

15 Fig. 6 illustrates a set of flow statistics  
corresponding to nine different flows (F1 through F9)  
received by a node implementing the AFFC mechanism of the  
present invention. The nine flows correspond to four  
classes Class 1 through Class 4. For purposes of  
20 explaining the invention, it will be assumed that the  
baselines shown in Fig. 4 represent the current flow rate  
baselines for the four classes.

Fig. 7 illustrates the exemplary results of  
25 applying the flow control methods of the invention to the  
nine flows illustrated in Fig. 6. The second from last  
row of Fig. 9 shows the flow throughputs for each of the  
nine flows after AFFC processing. Notice that if the  
congestion still continues, the AFFC flow regulation will

continue with the penalty ratio  $k$  on flow forwarding

rates--  $\lambda(t) = \frac{\lambda(t - \Delta t)}{k}$ .

In the case where the node is still saturated  
5 after application of the flow reduction techniques of the  
invention, additional packets may be dropped until the  
total volume of the flows equals the node's forwarding  
capacity. The dropping of such additional packets occurs  
in forwarding step 527 when necessary.

10

Flows F1 through F3 correspond to Class 1  
traffic, i.e., TCP/Web traffic directed to destination  
D2. From Fig. 4 it can be seen that the baseline flow  
rate for Class 1 traffic in the example is 1000 bits/s.

15

Flow F1 is found to be elastic responsive  
traffic having a flow arrival rate (800 bits/s) which is  
less than the baseline flow rate (1000 bits/s) for class  
1 traffic. Accordingly, no reduction is applied to the  
20 forwarding rate of flow F1.

Flow F2 is found to be elastic responsive  
traffic having a received flow rate (1200 bits/s) which  
is higher than the baseline flow rate (1000 bits/s) for  
25 class 1 traffic. Accordingly, forced flow rate  
reduction, e.g., dropping of packets, is applied to flow  
F2 to reduce its forwarding flow rate to the baseline of  
1000 bits/s.



Flow F3 is found to be elastic non-responsive traffic. Accordingly, the packets of Flow F3 are dropped, i.e., they are not forward.

5

Flows F4 and F5 correspond to Class 2 traffic, i.e., TCP/FTP traffic directed to destination D2. From Fig. 4 it can be seen that the baseline flow rate for Class 2 traffic in the example is 500 bits/s.

10

Flow F4 is found to be elastic non-responsive traffic. Accordingly, the packets of Flow F4 are dropped, i.e., they are not forward.

15

Flow F5 is found to be elastic responsive traffic having a received flow rate (400 bits/s) which is lower than the applicable baseline flow rate (500 bits/s) for class 2 traffic. Accordingly, no reduction is applied to the flow rate of flow F5.

20

Flows F6 thorough F9 correspond to best effort traffic, which does not have congestion control/avoidance scheme implemented in the protocol. Accordingly, responsiveness to congestion signals is not an issue with regard to these flows. Flows F6 and F7 correspond to Class 3 traffic, i.e., UDP/Echo traffic directed to destination D2. From Fig. 4 it can be seen that the baseline flow rate for Class 3 traffic in the example is 200 bits/sec. Flows F8 and F9 correspond to Class 4

25

traffic, i.e., UDP/DNS traffic directed to destination D2. From Fig. 4 it can be seen that the baseline flow rate for Class 4 traffic in the example is 100 bits/sec.

5               Flow F6 is found to be best effort traffic having a received flow rate (180 bits/s) which is lower than the applicable baseline flow rate (200 bits/s) for class 3 traffic. Accordingly, no reduction is applied to the flow rate of flow F6.

10              Flow F7 is found to be best effort traffic having an arrival rate (500 bits/s) which is higher than the baseline flow rate (200 bits/s) for class 3 traffic. Accordingly, forced flow rate reduction, e.g., dropping  
15 of packets, is applied to flow F7 to reduce its flow rate to the baseline flow rate of 2000 bits/s.

                Flow F8 is found to be best effort traffic having an arrival rate (200 bits/s) which is higher than  
20 the baseline flow rate (100 bits/s) for class 4 traffic. Accordingly, forced flow rate reduction, e.g., dropping of packets, is applied to flow F8 to reduce its flow rate to the applicable baseline flow rate of 100 bits/s.

25              Flow F9 is found to be best effort traffic having an arrival rate (90 bits/s) which is lower than the applicable baseline flow rate (100 bits/s) for class 4 traffic. Accordingly, no reduction is applied to the flow rate of flow F9.

Altogether, in the Fig. 7 example 2890 out of 5670 bits of data were dropped. The benefits of the AFFC method can be seen from this example.

5

As will now be discussed further benefits can be obtained by implementing Early Traffic Regulation (ETR) in accordance with the invention.

10

The ultimate purpose of ETR is to regulate flows which are responsible for congestion upstream of the node, e.g., a bottleneck node, where congestion is detected. By providing congestion control upstream of the point of congestion, greater protection against a collapse at the bottleneck due to congestion is provided than when congestion control is implemented solely at the bottleneck node.

15

Figure 9 illustrates the steps of the ETR method 900 of the present invention. The ETR method begins in step 902 with the execution of the ETR modules 228 in a plurality of network nodes including a destination node, bottleneck node and a node upstream of the point of congestion, e.g., the bottleneck node.

25

For purposes of explaining the ETR method of the present invention, consider the system of Figure 1 in which the method 900 may be implemented. It will be assumed for purposes of explanation that destination

device D1 110, node R7 127, and node R3 120 each include and implement the ETR module 228 of the present invention or at least a subset of the components thereof as will be discussed further below. It will also be assumed that

5 source device S1 102 is one of a plurality of source devices being used to flood device D1 110 with traffic. It will further be assumed that the flooding causes congestion at node R7 127, the bottleneck node thereby saturating the node for an extended period of time

10 sufficient to trigger ETR. From Fig. 1 it can be seen that node R3 is an upstream node relative to bottleneck node R7 127 on the path leading between source S1 102 and destination device D2.

15 Referring once again to the ETR method shown in Fig. 9, operation proceeds to step 904, wherein a node, e.g., the bottleneck node 127, detects congestion sufficient to trigger the ETR signaling to initiate flow control at an upstream node.

20 In response to detecting congestion at the bottleneck node 127 in step 904, operation proceeds to step 906 wherein the bottleneck node 127 sends a message to the destination node 110. The destination node 110 is

25 the node to which one or more of the packet flows that are causing the congestion, e.g., the non-responsive TCP flows and/or the aggressive allowable flows, are directed.

The receipt of the ETR message causes, in step 908, the destination node 110 to initiate a path back-tracing operation and to determine from a back-tracing operation the path of the packet flow or flows causing the congestion. The back tracing is performed using any one of a plurality of known techniques, which will not be discussed in any detail.

With the path of the flow or flows causing the congestion determined, in step 910 the destination node 110 sends the determined path information to the bottleneck node 127.

The path information obtained from the destination node 110 is used by the bottleneck node 127 in step 912. In step 912, the bottleneck node 110 sends an ETR signal, e.g., control message, to the upstream node 120 using the supplied path information. The ETR control message indicates whether flow control is to be applied (started) or discontinued (stopped) and includes information such as victims', e.g., targeted destination device IP address(es).

In step 914, in response to the ETR control message, the upstream node 120 implements flow rate control on the flow or flows identified in the ETR signaling message, e.g., the flows directed to the destination address indicated in the received ETR signaling message. The flow control techniques applied

may include, e.g., blocking of non-responsive elastic flows and limiting other flows as a function of the upstream node's baselines for flows of the type being subjected to flow control. Accordingly, aggressive flows  
5 subject to flow control will undergo forced reductions in their flow rates at the upstream node, e.g., they will be subject to having packets dropped.

The upstream node will continue to provide flow  
10 control until receiving an ETR message from the bottleneck node 127 indicating that the congestion problem is no longer present.

The ETR method 900 of the invention stops in  
15 step 916 when the nodes implementing the method are turned off.

The various subroutines and messaging used to implement the ETR method of the present invention will  
20 now be discussed in further detail with reference to Figs. 10-12.

Figure 10A illustrates an exemplary network node ETR module 228 which is implemented in software.  
25 The ETR module 228 includes a main ETR control routine 1030, and a plurality of subroutines. The subroutines include a Route-Receive subroutine (Rt-R) 1038, a Back-tracing message Send (Bt-S) subroutine 1034, an ETR Send

(ETR-S) subroutine 1036, and an ETR Receive (ETR-R) subroutine 1042.

Figure 10B illustrates an exemplary ETR module  
5 228' suitable for use in an end-host device, e.g., device 110. The module 228' includes Route-Send (Rt-S) subroutine 1032, and a Back-tracing message Receive (Bt-R) subroutine 1040.

10 Some or all of the sub-routines 1034, 1036, 1038, 1042 may be implemented in a network node. The purpose and various messages generated and/or processed by the various sub-routines will now be discussed with regard to Fig. 11. It also illustrates the sub-routines  
15 at each node 110, 127, 120 used to receive and/or send the messages communication between the illustrated nodes.

Fig. 11 illustrates messaging passed between destination node 110, bottleneck node 127 and upstream  
20 node 120 as part of the ETR method shown in Fig. 9.

In the destination node 110, two ETR subroutines are used for messaging. These are: (1) the Bt-R subroutine 1040, and (2) the Rt-S subroutine 1034.  
25 The Bt-R subroutine 1040 receives back tracing messages 1112 from upstream nodes requesting back tracing path reconstructed. In response to back tracking request messages 1112, the BT-R subroutine 1040 reconstructs back tracing path The Bt-R subroutine 1040, also receives back

tracing path messages 1110 sent from network node 127 with certain probability (e.g.  $1/20,000$ , which corresponds to one path being determined for every 20,000 packets of a given flow that are received).

5

The back tracing path message 1110 indicates information relating to the network node which sent it. The information may include the network node's IP address, the IP address for the previous network node in the path, and next network node's IP address. From the received back tracing path message 1110 the Bt-R subroutine 1040 estimates Round Trip Time (RTT) based on a timestamp record "*timestamp<sub>1</sub>*" located in the back tracing path message 1110.

15

The timestamp record marks the time the backtracing path message leaves each node identified in the backtracking path message. For example, the back tracing path message 1110 provides a reconstructed network-node chain originating from the node 127.

20

Similarly, 1126 is generated by node 120 sent to destination 110, 1124 is generated by an upstream network node. Combining  $n$  such messages received by the end-host 110, the following timestamp info can be collected:  $\{R_1(\text{timestamp}_1), R_2(\text{timestamp}_2), \dots, R_i(\text{timestamp}_i), \dots, R_n(\text{timestamp}_n)\}$ .

25



The  $RTT_i$  from anyone of the routers listed in the message 1110 will be:

$RTT_i = 2 \times (\text{TheTimeTheMessageWasReceived} - \text{timestamp}_i)$ ,

then the RTT of the network-node chain path can be

5 estimated by  $RTT \approx \max\{RTT_i | i \in [1, n]\}$ . Then the RTT from the sender, source 102, to the receiver, destination device 110,  $RTT_0$ , is estimated by  $RTT_0 = k \times RTT$ . The factor  $k$  may be preset or can be determined and/or estimated by the destination node.

10

The relationship between RTT 1205, back tracing path messages and the estimated round trip time period  $RTT_0$  1207 can be seen in Figure 12 for an exemplary communications path having a source 1202 and destination device 1214. In Fig. 12, the exemplary network node path-chain comprises, in addition to source and destination nodes 1202, 1212, respectively, nodes 1204, 1206, 1208, 1210 and 1212.

20

In accordance with one feature of the present invention estimated  $RTT_0$  time 1207 is conveyed to upstream nodes and is used to determine ETR signaling frequency. TCP-adaptive flows need at least one  $RTT_0$  time period 1207 to respond to congestion signals.

25

Accordingly, in some embodiments the period between ETR control signals is made as long or longer than one  $RTT_0$  time period 1207 so that the impact of TCP adaptive flow control mechanism, alone or in conjunction with applied flow control, can be properly gauged.

Referring once again to the Fig. 11 example,  
the Rt-S subroutine 1032 in the destination node 110  
receives path information 1110, 1126, 1124 and so forth  
5 by the Bt-R subroutine 1040 which reconstructs the  
tracing path. The reconstructed path is then sent as a  
message 1114 to the bottleneck node 127.

In the bottleneck node 127, there are three  
10 subroutines in addition to the main ETR routine 1030. As  
discussed above, the bottleneck node 127 requests route  
information from end hosts when potential flooding N-DoS  
congestion collapse is detected. In the bottleneck node,  
when congestion sufficient to merit early traffic  
15 regulation is detected, the main ETR control routine 1030  
is responsible for triggering, via control signal 1135,  
the sending of a backtracking request message.

Three sub-routines used in the bottleneck node  
20 127 are: (1) the Bt-S sub-routine 1034, (2) the Rt-R  
subroutine 1038, and (3) the ETR-S subroutine 1036.

The Bt-S subroutine 1034 sends path tracing  
information, e.g., path messages 1110, to downstream  
25 nodes. The path tracing messages may include path and  
timing information received by the Bt-S subroutine 1034  
from upstream nodes. The Bt-S 1034 is also responsible  
for sending back tracing initiation request messages 1112  
to the destination to request the reconstructed back

tracing path when triggered by the control routine 1030 via the signal 1135.

5 The Rt-R subroutine 1038 receives reconstructed tracing-path information 1114 from one or more end hosts, e.g., destination node 110. This information identifies upstream nodes to which ETR control messages may be sent. The path information is supplied to the ETR-S subroutine 1036. The ETR-S subroutine 1036 sends ETR control  
10 messages used to trigger flow control operations to one or more prior, i.e., upstream, network nodes along the reconstructed tracing path. The ETR control messages may include, e.g., upstream path information, and information identifying flows to be subject to flow control. Flows  
15 which are to be subject to flow control may be identified in an ETR control message by destination and/or one or more other identifiers.

As mentioned above, ETR control message sending  
20 rates, e.g., to enable/disable flow control in upstream nodes are bound to RTT estimated in end hosts based on the timestamp data attached in back tracing messages. Thus, time spacing between ETR control messages will be equal to, or greater than, the estimated RTT0 time  
25 period.

In the upstream node 120, in addition to the main ETR control routine 1030, there are three

subroutines: (1) the Bt-S subroutine 1034, (2) the ETR-R 1042, and (3) and the ETR-S subroutine 1036.

5       The Bt-S subroutine 1034 sends path tracing information in the form of message 1126 to the destination end-node.

10       The ETR-R subroutine 1042 in the upstream node 120 responds to ETR control messages received from downstream nodes, e.g., node 127. As part of the response to an ETR signal the main ETR control routine 1030 is triggered to initiate flow control on the flow or flows identified by information in the received ETR control message. In addition to initiating flow control  
15       in the current node, the main ETR control routine 1030 passes an ETR control message to the next upstream node, if any, identified in the received ETR message. The passed ETR control message is transmitted via the ETR-S subroutine 1036 which is responsible for transmitting ETR  
20       signals 1118 and 1122 to the nodes 120 and 127. As a result of ETR signal passing, flow control may be triggered on the identified flows in multiple upstream nodes.

25       In the above described manner AFFC flow control may be activated in one or more nodes preceding a bottleneck node, thereby eliminating some of the traffic directed to the bottleneck node. This upstream

protection reduces the risk of congestion collapse at the bottleneck node.

Numerous variations on the above described  
5 methods and apparatus are possible without departing from  
the scope of the invention.

11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816  
1817  
1818  
1819  
1820  
1821  
1822  
1823  
1824  
1825  
1826  
1827  
1828  
1829  
1830  
1831  
1832  
1833  
1834  
1835  
1836  
1837  
1838  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1860  
1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
2030  
2031  
2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039  
2040  
2041  
2042  
2043  
2044  
2045  
2046  
2047  
2048  
2049  
2050  
2051  
2052  
2053  
2054  
2055  
2056  
2057  
2058  
2059  
2060  
2061  
2062  
2063  
2064  
2065  
2066  
2067  
2068  
2069  
2070  
2071  
2072  
2073  
2074  
2075  
2076  
2077  
2078  
2079  
2080  
2081  
2082  
2083  
2084  
2085  
2086  
2087  
2088  
2089  
2090  
2091  
2092  
2093  
2094  
2095  
2096  
2097  
2098  
2099  
2100  
2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115  
2116  
2117  
2118  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159  
2160  
2161  
2162  
2163  
2164  
2165  
2166  
2167  
2168  
2169  
2170  
2171  
2172  
2173  
2174  
2175  
2176  
2177  
2178  
2179  
2180  
2181  
2182  
2183  
2184  
2185  
2186  
2187  
2188  
2189  
2190  
2191  
2192  
2193  
2194  
2195  
2196  
2197  
2198  
2199  
2200  
2201  
2202  
2203  
2204  
2205  
2206  
2207  
2208  
2209  
2210  
2211  
2212  
2213  
2214  
2215  
2216  
2217  
2218  
2219  
2220  
2221  
2222  
2223  
2224  
2225  
2226